Remarks

<u>Status of application</u>

Claims 1-99 were last examined and rejected in a Final Rejection dated December 8, 2008. The claims have been amended to further clarify Applicant's invention. Reexamination and reconsideration are respectfully requested. In view of the amendments made herein, the Request for Continued Examination filed herewith, the detailed discussion of the prior art references in Applicant's previously filed Request for Reconsideration (filed March 9, 2009), and the following remarks, reexamination and reconsideration are respectfully requested.

<u>Prior art rejections</u>

Claims 37-70 claims stand rejected under 35 U.S.C. 103(a) based on the combination of Newman (US Patent 7,266,699) and Lei (US Publication 2004/0255133 A1); claims 1-36 and 71-99 stand rejected under 35 U.S.C. 103(a) based on the combination of Newman (above) in view of Sato (US Patent 7,093,137 B1) and further in view of Lei (above). Applicant appreciates the Examiner's courtesy in discussing Applicant's claims and the prior art references on May 7, 2009. Based on that discussion, Applicant has amended the claims to further clarify the distinctive features of Applicant's invention so as to clearly distinguish the claimed invention from the prior art of record.

As discussed in detail in Applicant's previously filed Request for Reconsideration (incorporated herein by reference), Applicant's claimed invention is directed to SQL extensions that support creation and management of named encryption keys, thereby providing improved column-based encryption for database systems. Here, the term "named" key, as described in Applicant's specification and used in Applicant's claims, means that the key can actually be referenced in a <u>syntactically correct manner</u> within SQL statements provided by users. This approach, which differs substantially from prior art approaches, is accomplished by creating the named encryption key <u>with a syntactically unique name that can be parsed from within other SQL statements employing Applicant's SQL extensions</u>.

To the extent that that feature of Applicant's claimed invention was not clear, Applicant has amended all independent claims to explicitly recite the feature. For

18

example, claim 1 as amended herein recites:

parsing the first SQL statement, including **creating said named encryption key with a syntactically unique name that can be parsed from within other SQL statements employing said SQL extensions;**

(Emphasis added.)

Use of the syntactically unique name in a subsequent SQL statement to reference the named encryption key is also explicitly set forth in the amended claim:

parsing the second SQL statement, including **identifying said named encryption key upon parsing a portion of the statement that comprises the syntactically unique name for the key;**

(Emphasis added.)

(Applicant's other independent claims have been amended in a like manner.) Additional (and very specific) use of named encryption keys in SQL statements is provided in Applicant's dependent claims. In claim 9, for example, the named encryption key (*keyname*) is set forth as a syntactic construct used within an SQL CREATE TABLE statement.

As previously explained by Applicant, a named key approach has important practical implications for database encryption. The named encryption key that has been created may be referenced (i.e., used) in other SQL statements as a syntactical construct (i.e., as a programmatically accessible database object), such as in CREATE TABLE and ALTER TABLE statements (the subject matter in several claims, including for example 8, 9, 11, 12, 44, 45, 47, 48, 83 and 84), as well as other key-specific statements such as GRANT ALTER ENCRYPTION KEY. Here, one may programmatically reference directly within SQL statements that are performing a database task (e.g., creating tables) a particular named key as the encryption key used for column encryption of various

columns that are to be encrypted. Even the tasks of creating and managing the encryption key itself are achieved programmatically <u>entirely</u> through SQL statements. This approach greatly simplifies the creation and management of encryption keys, since the database user can perform **all** encryption-related tasks -- including management of the encryption keys themselves -- through familiar SQL statements <u>without</u> requiring any new or special client software. This is very different from anything described by the prior art of record.

The cited prior art contains absolutely no description whatsoever pertaining to use of SQL extensions to create a named encryption key having a syntactically unique name that allows the column encryption key to be programmatically accessed (for a multitude of purposes!) directly within SQL statements. Instead, those systems treat encryption keys as more or less an afterthought and assume that they can be obtained from some external source (e.g., the "master keys" Lei assumes are available). Those prior art systems lose an important opportunity to improve the efficiency and operation of database encryption.

For the reasons stated, it is respectfully submitted that Applicant's named encryption key approach -- where an encryption key is created with a syntactically unique name that allows the key to be accessed programmatically directly within SQL statements -- is not shown in the prior art and thus represents a patentable advance in the area of database encryption and security. Accordingly, it is believed that the claims are patentable under Section 103.

Any dependent claims not explicitly discussed are believed to be allowable by virtue of dependency from Applicant's independent claims, as discussed in detail above.

Conclusion

In view of the foregoing remarks and the amendment to the claims, it is believed that all claims are now in condition for allowance. Hence, it is respectfully requested that the application be passed to issue at an early date.

If for any reason the Examiner feels that a telephone conference would in any way expedite prosecution of the subject application, the Examiner is invited to telephone the undersigned at 408 884 1507.

Respectfully submitted,

Date: June 15, 2009

/John A. Smart/

John A. Smart; Reg. No. 34,929
Attorney of Record

408 884 1507
815 572 8299 FAX